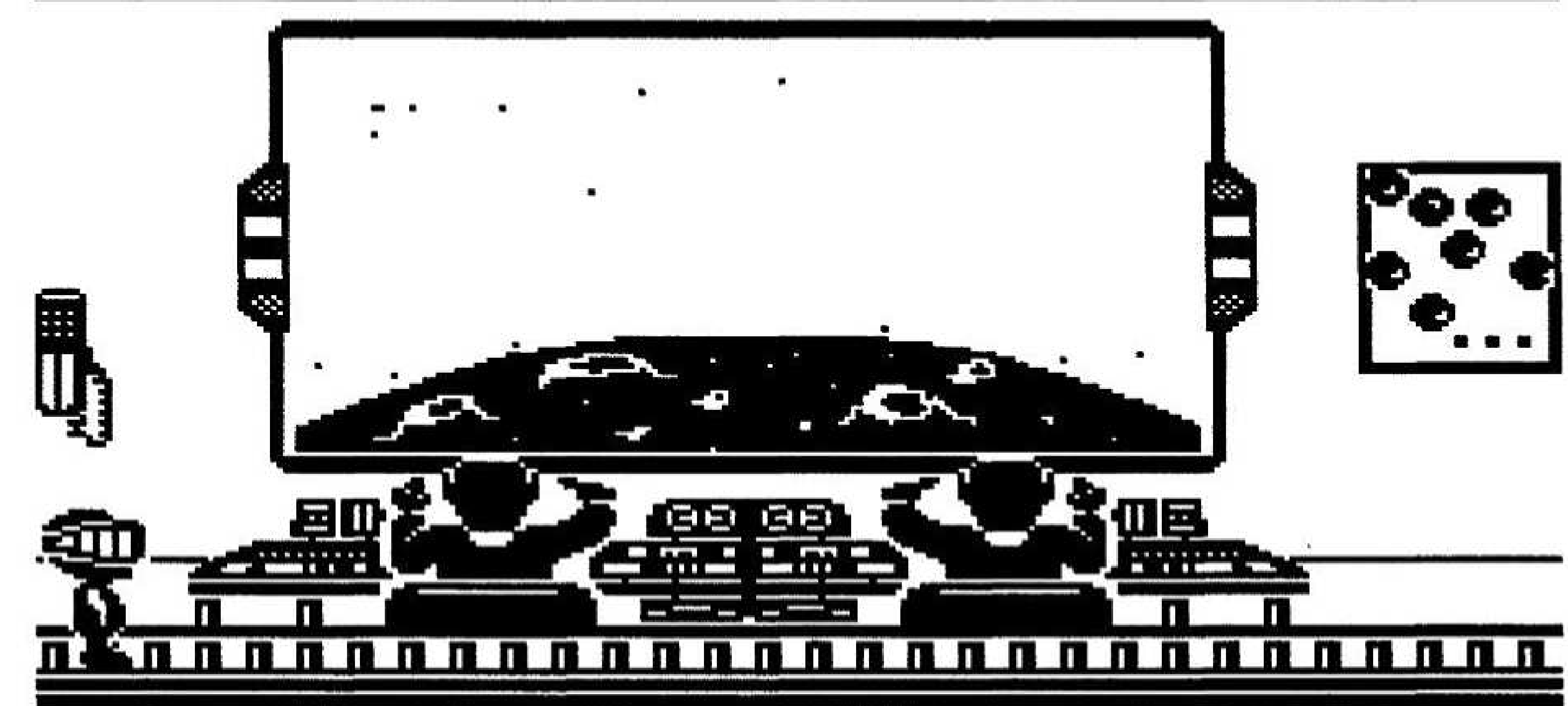
Spectrum Profi Club

für alle Spectrum und SAM Freunde



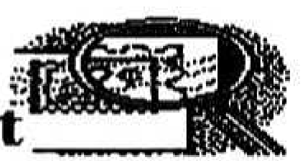
"The Magic Knight" (auf der Suche nach der längsten Praline im Universum): Wird er jemals wiederkommen?

Wolfgang & Monika Haller, Tel. 0221/685946
Im Tannenforst 10, 51069 Köln
Bankverbindung: Dellbrücker Volksbank
BLZ 370 604 26, Konto-Nr. 7404 172 012

Ausgabe 101

Mai 1998

PENG oder der Brief, der einen Knall hat



Im allgemeinen ist es nicht unsere Art, persönliche Briefe im Info zu veröffentlichen. Der nachfolgende jedoch hat es verdient, in die Annalen des SPC einzugehen. Damit auch ihr seht, womit man es manchmal zu tun bekommt, haben wir uns entschlossen, den Brief auch in seiner rechtschreiblichen Form abzudrucken. Eine Antwort darauf haben wir uns erspart.

Betreff: DEMO PENG Nr. 2
Bezug: Deine Post per 2.11.97

Hallo Wolfgang !

Leider komme ich aus Zeitlichen Gründen erst heute kurz dazu, zu schreiben.
Sag mal, willst Du mich verarschen ????
Was hat das Lügen-DEMO mit Humor zu tun ???
Wenn Du weiter mir so einen Schwachsinn schreibst, möchte ich von Dir ein psychiatrisches Gutachten, ob Du geistig noch ganz Dicht bist !!!!!
Hat Dich der minderwertige Frank Maurer angesteckt ??? Bist Du 8 Jahre alt ???
Bevor ich weitere Schritte einleite, setze ich Dir hiermit eine Frist bis zum 14.5.98, mir mitzuteilen, wer der Programmierer von der DEMO ist. Mit Adresse !!! Sonst gebe ich Deine Adresse als Programmierer an.
Sag mal, hast Du den Scheiß überhaupt mal gelesen und gesehen ???
Erstens ist die DEMO illegal, da ich nicht um Erlaubnis gefragt wurde, meinen Namen zu benützten !!!
Auch stimmen die Zeugenaussagen, gar nicht zu den Zeugenausagen, der Polizei überein !!!
So etwas nennt man Verarschung der Polizei, im Volksmund !
Auch untersage ich Dir hiermit, weiterhin die DEMO zu vertreiben und zu werben. Zuwiederhandlung werden mit einem Streitwert von 50 000 DM. geahndet !
Von jeder verkauften DEMO fordere ich 5 DM. pro Stück. Schmerzensgeld außergerichtlich !

Mit freundlichen Grüßen
Richard Raddatz

Mal schauen, was draus wird. Abgesehen von einer persönlichen Beleidigung und der Androhung einer Verleumdung finde ich es viel schlimmer, jemanden als minderwertig zu bezeichnen (auch wenn der Name falsch geschrieben ist).
Und noch was: Demos, egal welche und woher, werden von uns und auch von anderen Clubs als PD behandelt - und somit frei verteilt. Davon werde ich auch in diesem Fall nicht abweichen.

Wo vom WoMo-Team

Aus dem Internet gezogen...

Am Beispiel des Toasters wird im Internet die Benutzerfreundlichkeit der Produkte weltweit führender High-Tech-Unternehmen vorgestellt:

IBM: Wenn IBM Toaster herstellen würde, dann würden sie riesengroße Toaster bauen, zu denen die Leute das Brot hinbringen müßten, damit es über Nacht getoastet werden könnte. IBM würde den weltweiten Markt für sich in Anspruch nehmen und fünf oder sechs solcher Toaster in Einsatz bringen.
XEROX: Wenn Xerox Toaster herstellen würde, dann könnte man entweder ein- oder zweiseitig toasten, und alle nachfolgenden Scheiben würden heller und heller werden. Aber der Toaster würde das Brot auch für sie pressen.
ORACLE: Wenn Oracle Toaster herstellen würde, dann würden sie behaupten, ihr Toaster wäre mit allen Marken und Arten von Brot kompatibel, aber wenn Sie ihn zu Hause haben, werden Sie feststellen, daß die Bagle Maschine noch in der Entwicklung steckt, die Croissant-Extension in drei Jahren kommen wird, und daß das gesamte Gerät nur Rauch erzeugt.
SUN: Wenn Sun Toaster herstellen würde, dann würde der Toast oft verbrennen, aber Sie würden eine wirklich gute Tasse Java bekommen.
MICROSOFT: Wenn Microsoft Toaster herstellen würde, müßten Sie jedesmal, wenn Sie einen Laib Brot kaufen, einen Toaster erstehen. Sie müßten den Toaster nicht nehmen, aber Sie müßten ihn in jedem Falle bezahlen. Toaster '95 würde 15000 Pfund wiegen (dadurch benötigt er einen verstärkten Stahlboden), soviel Elektrizität verbrauchen, wie für die Versorgung einer kleinen Stadt nötig ist und 95 Prozent des Platzes in Ihrer Küche einnehmen. Microsoft würde behaupten, dies sei der erste Toaster, der Ihnen die Möglichkeit gibt zu kontrollieren, wie hell oder dunkel Ihr Toast sein soll, und würde heimlich Ihre anderen Küchengeräte befragen um herauszufinden, wer sie hergestellt hat.
SAP: Wenn SAP Toaster herstellen würde, wäre das Bedienungshandbuch ca. 10000 Seiten dick. Der Toaster hätte 2500 Schalter, die alle nach exaktem Muster und in genauer Reihenfolge eingeschaltet werden müßten. Ein Team von Basis- und Funktionsunternehmern würde ungefähr ein Jahr brauchen, um den Toaster bestmöglich zu konfigurieren, und dann nochmals sechs Monate, um ihn zu testen. In der Zwischenzeit müßte Ihre gesamte Familie ausgedehnte Ausbildungskurse besuchen, um zu lernen, wie der Toaster zu bedienen ist. Und wenn einmal alles läuft, so werden sie sagen, dann haben Sie den besten Toaster der Welt bekommen.

Eingesandt von Paul Webranitz
Borgasse 14, 54538 Kinheim

DIE PLUS D ECKE TEIL 7

Hallo Plus D Freunde,

wer nach Teil 6 der +D Ecke geglaubt hat, ich würde mich zur Ruhe setzen und Euch nicht mehr mit meinen Artikeln nerven, der hat sich getäuscht. Wenn mir vorher nicht der Himmel auf den Kopf fällt, wird es auch noch einen Teil 8 geben.
Heute gibt es jedenfalls etwas sehr spezielles. Im Teil 3 der +D Ecke habe ich Euch BETA DOS vorgestellt. Ich hoffe, daß viele von Euch +D Usern inzwischen mit BETA DOS arbeiten. Denn sonst wäre das, was ich mir heute "antun" werde ziemlich sinnlos. Wovon ich überhaupt rede?
Ich habe darauf hingewiesen, daß BETA DOS einige Fehler hat. Bevor Ihr weiter lest, schaut Ihr Euch am besten noch mal den Teil 3 der +D Ecke an. Insbesondere den Teil "BUGS". Dort habe ich Euch "BETA DOS BUG FIX" von Miles Kinloch empfohlen.
Wie gesagt gehört zu diesem Patch auch ein Text-File. Leider ist er in Englisch. Heinz Schober aus Dresden war so nett diesen Text ins deutsche zu übersetzen. Vielen Dank Heinz!!!
Bleibt für mich die Aufgabe alles abzutippen. Wolfgang kennt das Problem: Heinz schickt (fast) alles handschriftlich! Heinz, sag mal hast Du eigentlich keine Textverarbeitung??? Grausame Welt!!!
Aber wenn ich mich recht erinnere, habe ich versprochen diese Arbeit zu übernehmen. Außerdem soll Heinz seine Arbeit nicht umsonst gewesen sein. Vielleicht sollte ich hier nicht soviel Zeug erzählen, sondern einfach mal anfangen:

Englischer Originaltext: Miles Kinloch, Edinburgh.
Deutsche Übersetzung: Heinz Schober, Dresden.
Getippt, mit Bemerkungen versehen und teilweise überarbeitet: Guido Schell, Löhne.

BETA DOS BUG FIXES/ENHANCEMENTS

(BETA DOS Fehlerberichtigungen und Erweiterungen)

Das Programm "BETAFIX" beseitigt die folgenden Fehler und bringt einige neue Vorteile.

FEHLER

1. OPEN # Befehl.
Wenn man am Ende das IN oder OUT unterläßt, nimmt das DOS die Syntax nicht an, wenn in der selben Zeile noch eine Anweisung folgt, z.B. wenn ein Doppelpunkt der nächste Character nach dem Befehl ist. (Dieser Fehler betrifft auch G+DOS.)

2. FORMAT Befehl
Hier gibt es zwei Fehler:
a) Der spezielle Fehler FORMAT d1,n etc. ignoriert das angegebene Laufwerk und formatiert das zuletzt benutzte Laufwerk.

b) Der Befehl FORMAT d* wird von der Syntax nicht zugelassen.

3. Command codes (Befehls-Codes)
Diese sind vielleicht die schwerwiegendsten aller BETA DOS-Fehler, weil sie manche kommerzielle Programme beeinflussen, die das DOS vom Maschinencode her aufrufen. Wenn die Command Codes geändert sind, wird der größte Teil der Software ohne Probleme laufen. Ein paar Beispiele wären die +D konvertierte Version von Tasword 128, Hackers Workbench (S.D. Software), Profile (Glensoft), Disc-Kit und einige Utilitys von "Betterbytes". Die sechs folgenden codes sind fehlerhaft:

a) code 52 und 53.
Das Problem ist, daß das alternierende (alternate) HL-Registerpaar zerstört wird, was einen Absturz beim Rücksprung ins BASIC zur Folge hat, wenn ein USR-CALL gemacht wird, der diese command codes benutzt.

b) code 56.
Die Routine die durch RST 8 DB 56 aufgerufen wird, wurde nicht gegenüber der im PLUS D-ROM geändert, die auch G+DOS benutzt. Das ist zwar für Beta Dos gut, aber nicht bei Disketten die für mehr als 80 Files formatiert wurden; der Inhalt solcher Disketten würde immer zerstört werden. (code 56 dient zum Abschließen eines Files vom Maschinencode her, d.h. um einen Eintrag in das Directory zu schaffen.)

c) code 67.
Das ist der code um CAT für eine Diskette auszuführen und ist fatal fehlerhaft. Jeder Versuch ihn zu benutzen, bringt den Spectrum zum Absturz und zerstört möglicherweise gleichzeitig das DOS.

d) code 68 und 69.
Der entsprechende zu SAVE@ und LOAD@. Diese beiden bringen einen DOS Error, ohne Rücksicht darauf, ob ein wirklicher Fehler existiert. Siehe auch unter Punkt 4.

4. LOAD@/SAVE@ Kommando (und die entsprechenden command codes)
Wenn während LOAD@ oder SAVE@ ein Fehler gewesen ist, z.B. eine schreibgeschützte Diskette etc., ergibt das einen falschen Variablenwert in einer der DOS Systemvariablen. Wenn man dann RUN eingibt, um das DOS neu zu laden erfolgt sofort eine "End of File" Meldung. Auch wenn man versucht einen Snapshot zu machen, wird er falsch gesaved (gesichert). Zusätzlich kann es zur Zerstörung weiterer Files auf der Diskette kommen. Die Eingabe von CLEAR # setzt die Variable zurück. Wenn aber beispielweise zur gleichen Zeit einige Kanäle (Streams) offen sind oder wenn der Fehler durch die command codes gekommen ist und ein Einsprung ins BASIC unmöglich war, ist das sicher keine ideale Lösung des Problems.

5. SAVE@ Kommando.
Da gibt es ein Problem mit der Routine, die den Floppymotor extra eine Sekunde vorher drehen läßt, damit er genügend hohe Drehzahl hat bevor ein Sektor beschrieben wird. Das System ignoriert auf welche Floppy das SAVE@ stattfinden soll. Es wird stattdessen die zuletzt benutzte Floppy angesprochen. Das läuft gut, wenn man jeweils eine Diskette in beiden Laufwerken hat. Wenn in der zuletzt benutzten Floppy keine Disk eingelegt ist, läuft der Motor dieser Floppy undefiniert weiter, bis eine Diskette eingeschoben wird.

6. Die Benutzung des Großbuchstaben "D" bei SAVE und ERASE.
Wenn es auch manchmal so scheint, ist der automatische CAT, das beim G+DOS bei diesen Kommandos stattfindet, bei BETA DOS doch nicht ganz beseitigt. Der CAT erfolgt, aber man kann es nicht sehen, weil der STREAM so umgeleitet wird, daß der CAT im Arbeitsraum (workspace) des BASIC erfolgt, anstatt auf dem oberen Bildschirm (screen). Dabei werden keine Überprüfungen gemacht, welcher Platz für den Arbeitsraum zur Verfügung steht etc. Das kann manchmal Probleme machen, besonders wenn vorher ein INPUT gemacht wurde, der ja auch den Arbeitsraum benutzt. Beispiel von Guido:

```
10 INPUT a
20 SAVE D1 "TEST"
RUN [ENTER].
Das System stürzt ab.
```

7. MOVE Kommando.
Die Form MOVE #s TO d1"name" (nicht in der +D Anleitung dokumentiert) hat zwei schwerwiegende Fehler. Der erste ist, daß der Directory Eintrag für ein neues File nicht im Katalog sichtbar wird, obwohl er existiert. Das zweite Problem ist, daß bei dieser Form des MOVE Kommandos, wenn der File-Name schon benutzt wurde und man "N" bei der OVERWRITE Abfrage eingegeben hat, sich später der Spectrum bei der Eingabe von CLEAR # oder CLOSE #• "aufhängt". Das ist so, weil einige Zeiger (pointers) nicht zurückgesetzt wurden und CLEAR # und CLOSE #• dann desorientiert sind, wieviel Speicherplatz sie zu fordern haben (Betrifft auch G+DOS.)

8. BETA DOS FN calls, z.B. DEF FN s(x) = USR 8
Möglicherweise könnte es hier ein Problem mit dem 128er Editor geben, wenn solche Calls als direkte Befehle eingegeben werden, besonders diejenigen die OPENTYPE-Files betreffen, z.B. FN e, FN 1 und FN p. Der 128K Editor ist mit dem Inhalt des IX-Registerpaares ziemlich beschäftigt (genauer gesagt mit dem (IX+1), wenn nicht Null), wenn er mit einer Zeilenausgabe zu tun hat. Die letzteren drei Funktionen benutzen IX für sich allein, so daß für 100% Sicherheit dieses Register danach korrekt gesetzt sein muß. Weitere Details zu diesem IX Problem sind im Artikel "Plus D Alters" im Format Magazin Ausgabe März 1988 (Vol.1, No.8) zu finden. Dieses war ja auch etwas, was das vorangegangene DOS beeinflußte (D.h. vor DOS Version G+DOS 2a).

9. Das Band (Tape) SAVE Kommando
Wenn bei angeschlossenem +D ein Null-File-Name (SAVE "") oder einer länger als 10 Zeichen eingegeben wurde, bekommt man die Fehlermeldung "Nonsense in BASIC" anstatt "Invalid File name". Das ist allerdings nur ein kleiner Fehler.
(Anmerkung von Guido Schell: Ich habe es mal ausprobiert, bei angeschlossenem +D ohne geladenes DOS funktioniert alles richtig. Bei geladenem BETA DOS und SAVE "" erscheint tatsächlich "Nonsense in BASIC". Aber bei SAVE "12345678900" erscheint bei mir "Invalid Device" und bei SAVE "sssssssssss" (11 x s) erscheint bei mir "Variable not found"! Wenn ich ein mit BETAFIX gepatchtes BETA DOS verwende, erscheint in beiden Fällen so wie es richtig ist "Invalid file name". G+DOS habe ich nicht probiert.)

10. "Drucken" in einen random access files. Hier gibt es zwei Fehler:
a) Im BETA DOS Handbuch steht, daß die Text- oder Kontroll-Charactere nicht beachtet werden, die vor der eingegebenen Variable eingegeben werden, wenn man vor einem mit RND eröffneten File einen INPUT macht. Z.B. INPUT #4; "Text"; a$. Das trifft aber nur für den 48K Modus zu. Im 128K BASIC wird ein solcher Text ohne Beanstandung geschrieben.
b) Die Benutzung von Farben in einem PRINT Kommando, das Daten zu dem File sendet, können einen Fehler verursachen. Eingebettete Control Codes sind in Ordnung, aber wenn man PAPER-, INK- etc. Tokens im PRINT Befehl benutzt, z.B. PRINT #4; INK 5; "text", kann das eine "Nonsense in BASIC" Fehlermeldung zur Folge haben.

VERBESSERUNGEN

1. Ein neues Kommando
Man kann jetzt mit LOAD dn oder SAVE dn (n = Nummer des Laufwerkes, also 1 oder 2) das Laufwerk wechseln. Das kann z.B. vor Verwendung das Kommandos LOAD pn nützlich sein.

2. SAVE ... TO ...
Diese Kommando wurde hinsichtlich des Kopierens zwischen zwei Laufwerken verbessert. Das +D nimmt nur kurze Zeit den Inhalt des Track Registers eines Laufwerkes wahr (Laufwerk-Kopf-Position). Das kann manchmal zu Problemen führen, besonders wenn lange Files sowie 128K Snapshots kopiert werden, welche nicht in nur einem Durchgang kopiert werden können. Das System kann über die wahre Position des Kopfes "verwirrt" sein und wenn der File auf die Tracks nahe am Ende der Disk kopiert wird kann alles durcheinander kommen. Das System stoppt mit einem Error. BETAFIX "flickt" die SAVE-TO Routine, so daß getrennte Meldungen für die Track Register der beiden Laufwerke genommen werden. Damit wird nicht nur eine größere Zuverlässigkeit, sondern auch eine höhere Geschwindigkeit erreicht (ca. 15%).

=== Ende der Übersetzung ===

Puuuuh, geschafft! Na, habt Ihr alles kapiert? Ich (zugegeben) ganz bestimmt nicht. Mangels Sach- und Englischkenntnisse, kann ich auch nicht beurteilen ob das geschriebene richtig ist. Wie oben bereits angedeutet, habe ich mir die Freiheit genommen (wo es mir sinnvoll erschien und wo es meine Englischkenntnisse zuließen) den Text von Heinz ein wenig zu ändern. Wenn man den Original Text mit der deutschen Übersetzung vergleicht, kann man teilweise sehen wie schwierig es ist, so zu übersetzen, daß der Sinn erhalten bleibt. Das liegt vor allem an der Fachsprache im Computerbereich. Manche Dinge lassen sich schlecht "eindeutschen" und man sucht krampfhaft nach einer Umschreibung für irgendwas. Da ist es manchmal besser, man läßt das englische Original wie es ist.
Dann hoffe ich noch, daß beim Abdruck dieses Textes in den verschiedenen Publikationen das Sonderzeichen "@" der +D Syntax richtig gedruckt wird. Das war bisher nicht immer der Fall. Aber Ihr seid ja nicht auf den Kopf gefallen und habt die Fehler hoffentlich erkannt, oder? Gemeint ist das Sonderzeichen welches man mit SYMB SHIFT und Taste "2" erreicht. Der richtige Ausdruck ist abhängig vom eingestellten Zeichensatz der am Drucker eingestellt ist. Oder?
Abschließend möchte ich noch anmerken, daß ich nicht für die Richtigkeit der Aussagen von Mile's garantieren kann. Das gleiche gilt für die Übersetzung. Ich persönlich benutze nur noch das mit BETAFIX gepatchte BETA DOS und bin sehr zufrieden. Im Zweifelsfall den englischen Originaltext benutzen und (sofern möglich) das beschriebene einfach ausprobieren!
So, eigentlich wollte ich für heute erstmal Schluß machen, aber da flattert mir von Wolfgang Haller ein Brief ins Haus. Er hat mir einen weiteren von Heinz Schober übersetzten Text geschickt. Selbstverständlich auch handgeschrieben. Etwas davon will ich hier noch reinbringen weil es ganz gut hierher paßt:
In Teil 2 der +D Ecke habe ich Euch verschiedene Kopierprogramme vorgestellt. Mit dabei war auch BACKUP, ein Sektor-Kopierprogramm welches zum BETADOS Programmpaket gehört. Ich habe auch geschrieben, daß es damit manchmal Probleme geben kann. Miles Kinloch hat auf den Artikel mit folgenden Zeilen reagiert, die ebenfalls von Heinz Schober übersetzt wurden:

Obwohl ich dem deutschen Text nicht gut folgen konnte, erkannte ich, daß Probleme mit dem Beta Dos Backup angesprochen wurden. Dieses Programm war wirklich in seiner ursprünglichen Form fehlerhaft. Hauptsächlich durch 2 Fehler: Es war mit zwei Laufwerken bezüglich der Track-Register-Anomalie (IN/OUT 235) unzuverlässig und dann auch wenn man eine nahezu volle Disk kopiert, wurden die beiden letzten Sektoren oftmals nicht übertragen. Ich habe diese beiden Fehler berichtigt und ebenso die RAM DISK geändert: Die Benutzung der RAM DISK verringert die Zahl der Diskettenwechsel bei Verwendung von nur einem Laufwerk. Aber wenn zwei Laufwerke eingesetzt werden, ist der Einsatz der RAM DISK nicht notwendig, diese verlangsamt den Kopiervorgang sogar! Deshalb habe ich das durch eine Überbrückung in Systemen mit zwei Laufwerken geändert, um da die Kopierzeit um einige Sekunden zu verkürzen. Schließlich habe ich einen Test eingebaut, damit es mit einem 48K Spectrum arbeitet. Es sind aber viel mehr Diskettenwechsel erforderlich (z.B. bei Verwendung von nur einem Laufwerk), da die RAM DISK bei einem 48K Gerät nicht verfügbar ist. Die von mir überarbeitete Version von Beta Dos Backup ist bei "WoMo PD" erhältlich.

=== Ende der (sehr freien) Übersetzung ===

Den Originaltext findet Ihr übrigens im SPC Clubheft 9.96 auf Seite 15 am Ende seines Beitrags "Plus D Pitfalls" oder auch im SPC Heft 10.96 auf Seite 16 oben links.
Auch hier habe ich Heinz seine Übersetzung noch etwas geändert. Jetzt ist die Übersetzung zwar nicht 100% identisch mit dem Originaltext, aber mir scheint es wichtiger, daß der Text von der Sache her verstanden wird. Das ist sicher auch im Sinne von Miles und Heinz. Schließlich mache ich hier keinen Englisch Kursus sondern die +D Ecke!

Guido Schell, Auf dem Stocke 37
32584 Löhne, Telefon 05732 8769

DIE SEITEN FÜR DEN SAM!

VM - ein virtueller Z80 Prozessor?

Hallo SAM-Freunde!
Da es in letzter Zeit ziemlich ruhig um den SAM im Info geworden ist, und wir auf die 100. Ausgabe zugehen (ich hoffe, die 101. tut's auch. Wo), dachte ich mir, daß es mal wieder Zeit ist, etwas für den SAM zu schreiben.
Also habe ich mich hingesetzt und mir ein Thema ausgedacht. Das ist gar nicht so leicht, zumal mein SAM leider kaputt ist (Soundchip und eine Tastaturreihe) und der Emulator auf meinen Mac (zu alt!) nicht läuft. Daher ist alles was ich hier sage bzw. behaupte rein theoretisch. Wenn etwas nicht 100% stimmt oder ich Syntaxfehler mache, bitte entschuldigt.
Als der noch SAM noch lief und ich in Assembler programmierte, war es grausam sich um das Speichermanagement zu kümmern. Es waren immer nur 64Kb aufgrund des internen Aufbaus des Z80 des Speicher zugreifbar. Das Paging half zwar, nur mußte man beim Programmieren immer aufpassen, daß man die richtige Speicherbank einschaltet, und sein Programm nicht zu lang wurde, das es in den verrückbaren Speicherbereich hineinfiel. Kurzum, wurde das Programm länger oder der Speicherzugriff komplizierter, mußten richtig komplizierte Techniken entwickelt werden.
Eine Idee wurde geboren: Der Z80VM. Das soll ein neuer virtueller Z80-Prozessor werden. VM steht für "Virtual Machine" und bedeutet übersetzt nichts anderes als "Virtuelle Maschine". Das momentan berühmteste Beispiel für VM ist Java. Der Z80VM sieht auf den ersten Blick aus wie der handelsübliche Z80. Es ist nur ein erweiterter Z80, der 3 neue 21bit-Register und ca. 100 neue Befehle hat, die softwaremäßig emuliert werden. Daher ist dieser Z80 nur eine VM.
Nun wie soll dieser Z80VM meine Probleme lösen? Ganz einfach. Der Z80VM besitzt 3 neue Register: VX, VY und VO ("V" steht für virtuell), und sind 21bit-Register. Die neuen Befehle bewirken den Zugriff und Verwendung der neuen Register. Die Tabelle am Ende verdeutlicht die neuen Befehle mit Bedeutung und Parameter.
Warum sind die neuen Register 21bit? Verwirrend, nicht wahr? Nun, logisch müßten es nun 24bit sein, aber diese Register sollen mir das Speichermanagement erleichtern. Nun wie? Um 21bit im Speicher darzustellen sind 3 Bytes notwendig. Das erste Byte ist die Bank, und die letzen beiden Bytes zeigen auf die Adresse in der Bank. Eine Bank ist nur 16Kb groß, also hat einen Adressbereich von 0 bis 16383. Und die Adresse 16383 benötigt nur 13 Bits. Das macht zusammen 21 Bit. Es stehen damit die vollen 4 Mb (mit 4x1Mb Speichererweiterung) in 255 Bänken a 16kb zur Verfügung.
Außerdem bin ich fasziniert von der DLL-Technik (Windows) bzw. der SHARED LIBRARY-Technik (MacOs). Das brachte mich auf die Idee, daß es doch eigentlich egal sein sollte an welche Speicheradresse ich mein Programm lade. Es sollte immer laufen, ob ich es nach 32768 lade oder 237453. Dieses Problem will ich mit Hilfe des VO-Registers lösen. VO gibt sozusagen den Nullpunkt im Speicher an. Beispiel? Okay.
Nehmen wir an VO hat den Wert 36000. Nun sagen wir dem Z80VM: Springe relativ nach Adresse 49152 (JR 49152). Der Z80VM errechnet nun die absolute Speicheradresse (49152 + VO), die lautet 85152, und springt dorthin. Hätte VO nun den Wert 30000, würde der Z80VM zur absoluten Speicheradresse 79152 springen. Damit wäre unser Programm IMMER frei beweglich, sprich REALLOKIERBAR. Natürlich kann man auch weiterhin mit JP <pnn> absolut springen, in dem Falle direkt nach Speicheradresse 49152.
Ich weiß, daß das alles hier ziemlich kompliziert ist, aber ich denke mir doch, daß mich die Programmierer unter uns doch ein klein wenig verstanden haben. Wenn nicht dann emailt mir einfach. Darüber würde ich mich sehr freuen, auch wenn es nicht um den Z80VM geht. Genug geredet, jetzt lasse ich Beispiele für mich sprechen.

Beispiel: Eine Routine zum Addieren von 10000 und 20000, und dann zu der Adresse springen, die das Ergebnis anzeigt, also 30000.

```
LD   VX,10000 ; VX bekommt den Wert 10000
LD   VY,20000 ; VY bekommt den Wert 20000
ADD  VX,VY    ; VX und VY addieren, das
              ; Ergebnis in VX speichern
JP   (VX)     ; Zur Adresse springen, das VX
              ; anzeigt -> 30000
```

Ich hoffe, daß ich bald den Sourcecode für die VM fertig habe, so daß, falls sich jemand dafür interessiert, diesen bei mir anfordern kann.
So nun die Tabelle mit allen neuen Opcodes, die über RST 8 eingeleitet werden. Beispiel:

```
LD VX,32767 -> RST 8   ; Aktiviere VM
DB   13                ; Opcode für LD VX,<pnn>
DB   1                 ; Erste Byte ist Page <p>
                       ((=32767/16384)-1)
DW   10000             ; danach Adresse <nn>
                       (=32767-p*16384)
```

Opcode	Mnemonic	Bedeutung
0	RET	Ersatz für RET, damit VM richtig zurückspringen kann. Normaler RET kann nur innerhalb 0 und 65535 zurückspringen. Der RET von der VM kann zwischen 0 und 4194304.
1	JP <pnn>	Absoluter Sprung nach Page <p>, Adresse <nn>
2	JR <pnn>	Relativer Sprung von Offset V0 um Page <p>, Adresse <nn>
3	JP [VX]	Absoluter Sprung zur Adresse, die VX beinhaltet
4	JR [VX]	Relativer Sprung von Offset V0 um Wert, den VX beinhaltet
5	JP [VY]	Absoluter Sprung zur Adresse, die VY beinhaltet
6	JR [VY]	Relativer Sprung von Offset V0 um Wert, den VY beinhaltet
7	CALL <pnn>	Absoluter Sprung nach Page <p>, Adresse <nn> mit Rücksprung durch RET (Opcode 0)
8	CALL [VX]	Wie JP (VX) nur mit Rücksprung RET-Möglichkeit
9	CALL [VY]	Wie JP (VY) nur mit Rücksprung RET-Möglichkeit
10	CALR <pnn>	Relativer Sprung von Offset V0 um Page <p>, Adresse <nn> mit Rücksprung durch RET (Opcode 0)
11	CALR [VX]	Wie JR (VX) nur mit Rücksprung RET-Möglichkeit
12	CALR [VY]	Wie JR (VY) nur mit Rücksprung RET-Möglichkeit
13	LD VX,<pnn>	Lade Register VX mit 21bit Wert <p>=Page, <nn>=Adr.
14	LD VY,<pnn>	Lade Register VY mit 21bit Wert <p>=Page, <nn>=Adr.
15	LD V0,<pnn>	Lade Offset-Register V0 mit 21bit Wert
16	LD VXA,<p>	Lade Pagebyte des VX-Registers mit Page <p>
17	LD VYA,<p>	Lade Pagebyte des VY-Registers mit Page <p>
18	LD V0A,<p>	Lade Pagebyte des V0-Offset-Registers mit Page <p>
19	LD VXHL,<nn>	Lade Adressbytes des VX-Registers mit Adresse <nn>
20	LD VYHL,<nn>	Lade Adressbytes des VY-Registers mit Adresse <nn>
21	LD V0HL,<nn>	Lade Adressbytes des V0-Offset-Registers mit <nn>
22	LD VXH,<n>	Lade Highbyte der Adressbytes von VX mit <n>
23	LD VYH,<n>	Lade Highbyte der Adressbytes von VY mit <n>
24	LD V0H,<n>	Lade Highbyte der Adressbytes von V0 mit <n>
25	LD VXL,<n>	Lade Lowbyte der Adressbytes von VX mit <n>
26	LD VYL,<n>	Lade Lowbyte der Adressbytes von VY mit <n>
27	LD V0L,<n>	Lade Lowbyte der Adressbytes von V0 mit <n>
28	LD VX,[<pnn>]	Lade VX mit Wert von absoluter Speicheradr. <pnn>
29	LD VY,[<pnn>]	Lade VY mit Wert von absoluter Speicheradr. <pnn>
30	LD V0,[<pnn>]	Lade V0 mit Wert von absoluter Speicheradr. <pnn>
31	LR VX,[<pnn>]	Lade VX mit Wert von relativer Speicheradr. <pnn>
32	LR VY,[<pnn>]	Lade VY mit Wert von relativer Speicheradr. <pnn>
33	NOP	- Reserved -
34	LD VXA,[<pnn>]	Lade Pagebyte von VX mit Wert von absoluter Speicheradr. <pnn>
35	LD VYA,[<pnn>]	Lade Pagebyte von VY mit Wert von absoluter Speicheradr. <pnn>
36	LD V0A,[<pnn>]	Lade Pagebyte von V0 mit Wert von absoluter Speicheradr. <pnn>
37	LR VXA,[<pnn>]	Lade Pagebyte von VX mit Wert von relativer Speicheradr. <pnn>
38	LR VYA,[<pnn>]	Lade Pagebyte von VY mit Wert von relativer Speicheradr. <pnn>
39	NOP	- Reserved -
40	LD VXHL,[<pnn>]	Lade Adressbytes von VX mit Wert von absoluter Speicheradr. <pnn>
41	LD VYHL,[<pnn>]	Lade Adressbytes von VY mit Wert von absoluter absoluter Speicheradr. <pnn>
42	LD V0HL,[<pnn>]	Lade Adressbytes von V0 mit Wert von absoluter absoluter Speicheradr. <pnn>
43	LR VXHL,[<pnn>]	Lade Adressbytes von VX mit Wert von relativer Speicheradr. <pnn>
44	LR VYHL,[<pnn>]	Lade Adressbytes von VY mit Wert von relativer Speicheradr. <pnn>
45	NOP	- Reserved -
46	LD VXH,[<pnn>]	Lade Highbyte der Adressbytes von VX mit Wert von absoluter Speicheradr. <pnn>
47	LD VYH,[<pnn>]	Lade Highbyte der Adressbytes von VY mit Wert von absoluter Speicheradr. <pnn>
48	LD V0H,[<pnn>]	Lade Highbyte der Adressbytes von V0 mit Wert von absoluter Speicheradr. <pnn>
49	LR VXH,[<pnn>]	Lade Highbyte der Adressbytes von VX mit Wert von relativer Speicheradr. <pnn>

Opcode	Mnemonic	Bedeutung
50	LR VYH,[<pnn>]	Lade Highbyte der Adressbytes von VY mit Wert von relativer Speicher-adr. <pnn>
51	NOP	- Reserved -
52	LD VXL,[<pnn>]	Lade Lowbyte der Adressbytes von VX mit Wert von absoluter Speicher-adr. <pnn>
53	LD VYL,[<pnn>]	Lade Lowbyte der Adressbytes von VY mit Wert von absoluter Speicher-adr. <pnn>
54	LD V0L,[<pnn>]	Lade Lowbyte der Adressbytes von V0 mit Wert von absoluter Speicher-adr. <pnn>
55	LR VXL,[<pnn>]	Lade Lowbyte der Adressbytes von VX mit Wert von relativer Speicher-adr. <pnn>
56	LR VYL,[<pnn>]	Lade Lowbyte der Adressbytes von VY mit Wert von relativer Speicher-adr. <pnn>
57	NOP	- Reserved -
58	LD A,VXA	Lade Register A mit Pagebyte von VX
59	LD A,VYA	Lade Register A mit Pagebyte von VY
60	LD A,V0A	Lade Register A mit Pagebyte von V0
61	LD A,VXH	Lade Register A mit Highbyte der Adressbytes von VX
62	LD A,VYH	Lade Register A mit Highbyte der Adressbytes von VY
63	LD A,V0H	Lade Register A mit Highbyte der Adressbytes von V0
64	LD A,VXL	Lade Register A mit Lowbyte der Adressbytes von VX
65	LD A,VYL	Lade Register A mit Lowbyte der Adressbytes von VY
66	LD A,V0L	Lade Register A mit Lowbyte der Adressbytes von V0
67	LD VXA,A	Lade Pagebyte von VX mit Register A
68	LD VYA,A	Lade Pagebyte von VY mit Register A
69	LD V0A,A	Lade Pagebyte von V0 mit Register A
70	LD VXH,A	Lade Highbyte der Adressbytes von VX mit Register A
71	LD VYH,A	Lade Highbyte der Adressbytes von VY mit Register A
72	LD V0H,A	Lade Highbyte der Adressbytes von V0 mit Register A
73	LD VXL,A	Lade Lowbyte der Adressbytes von VX mit Register A
74	LD VYL,A	Lade Lowbyte der Adressbytes von VY mit Register A
75	LD V0L,A	Lade Lowbyte der Adressbytes von V0 mit Register A
76	LD HL,VXHL	Lade Register HL mit Adressbyte von VX
77	LD HL,VYHL	Lade Register HL mit Adressbyte von VY
78	LD HL,V0HL	Lade Register HL mit Adressbyte von V0
79	LD VXHL,HL	Lade Adressbyte von VX mit Register HL
80	LD VYHL,HL	Lade Adressbyte von VY mit Register HL
81	LD V0HL,HL	Lade Adressbyte von V0 mit Register HL
82	LD A,[<pnn>]	Lade Register A mit Wert in absoluter Spreicheradresse <pnn>
83	LD A,[VX]	Lade Register A mit Wert in absoluter Speicheradresse, die Register VX anzeigt
84	LD A,[VY]	Lade Register A mit Wert in absoluter Speicheradresse, die Register VY anzeigt
85	LD A,[V0]	Lade Register A mit Wert in absoluter Speicheradresse, die Register V0 anzeigt
86	LR A,[<pnn>]	Lade Register A mit Wert in relativer Speicheradresse <pnn>
87	LR A,[VX]	Lade Register A mit Wert in relativer Speicheradresse, die Register VX anzeigt
88	LR A,[VY]	Lade Register A mit Wert in relativer Speicheradresse, die Register VY anzeigt
89	NOP	- Reserved -
90	LD HL,[<pnn>]	Lade Register HL mit Wert in absoluter Speicheradresse <pnn>
91	LD HL,[VX]	Lade Register HL mit Wert in absoluter Speicheradresse, die Register VX anzeigt
92	LD HL,[VY]	Lade Register HL mit Wert in absoluter Speicheradresse, die Register VY anzeigt
93	LD HL,[V0]	Lade Register HL mit Wert in absoluter Speicheradresse, die Register V0 anzeigt
94	LR HL,[<pnn>]	Lade Register HL mit Wert in relativer Speicheradresse <pnn>
95	LR HL,[VX]	Lade Reg. HL mit Wert in relativer Speicheradresse, die Reg. VX anzeigt

Opcode	Mnemonic	Bedeutung
96	LR HL,(VY)	Lade Reg. HL mit Wert in relativer Speicheradresse, die Reg. VY anzeigt
97	NOP	- Reserved -
98	ADD A,VXA	Addiere Register A mit Pagebyte von VX
99	ADD A,VYA	Addiere Register A mit Pagebyte von VY
100	ADD A,VOA	Addiere Register A mit Pagebyte von VO
101	ADD A,VXH	Addiere Reg. A mit Highbyte der Adressbytes von VX
102	ADD A,VYH	Addiere Reg. A mit Highbyte der Adressbytes von VY
103	ADD A,VOH	Addiere Reg. A mit Highbyte der Adressbytes von VO
104	ADD A,VXL	Addiere Reg. A mit Highbyte der Adressbytes von VX
105	ADD A,VYL	Addiere Reg. A mit Highbyte der Adressbytes von VY
106	ADD A,VOL	Addiere Reg. A mit Highbyte der Adressbytes von VO
107	SUB VXA	Subtrahiere Pagebyte von VX von Register A
108	SUB VYA	Subtrahiere Pagebyte von VY von Register A
109	SUB VOA	Subtrahiere Pagebyte von VO von Register A
110	SUB VXH	Subtrahiere Highbyte der Adr.bytes von VX von Reg.A
111	SUB VYH	Subtrahiere Highbyte der Adr.bytes von VY von Reg.A
112	SUB VOH	Subtrahiere Highbyte der Adr.bytes von VO von Reg.A
113	SUB VXL	Subtrahiere Lowbyte der Adr.bytes von VX von Reg. A
114	SUB VYL	Subtrahiere Lowbyte der Adr.bytes von VY von Reg. A
115	SUB VOL	Subtrahiere Lowbyte der Adr.bytes von VO von Reg. A
116	ADD HL,VXHL	Addiere HL mit Adressbytes von VX
117	ADD HL,VYHL	Addiere HL mit Adressbytes von VY
118	ADD HL,VOHL	Addiere HL mit Adressbytes von VO
119	ADD VXHL,HL	Addiere Adressbytes von VX mit HL
120	ADD VYHL,HL	Addiere Adressbytes von VY mit HL
121	ADD VOHL,HL	Addiere Adressbytes von VO mit HL
122	SBC HL,VXHL	Subtrahiere Adressbytes von VX von HL
123	SBC HL,VYHL	Subtrahiere Adressbytes von VY von HL
124	SBC HL,VOHL	Subtrahiere Adressbytes von VO von HL
125	SBC VXHL,HL	Subtrahiere HL von Adressbytes von VX
126	SBC VYHL,HL	Subtrahiere HL von Adressbytes von VY
127	SBC VOHL,HL	Subtrahiere HL von Adressbytes von VO

Stephan Haller, Schützheider Weg 32, 51465 Bergisch Gladbach, Telefon: 02202/31052
Email: Nomad@change.gun.de -oder- cadoritzba@aol.com

FLASH mit Maus für Disk & Harddisk

Das arbeiten mit B-DOS und Harddisk ist schon Klasse. Wer wie ich auch eine Two-Drive-Flashversion mit Maus benutzen möchte (eine echte Aufwertung), der sollte seine entsprechende Version im Basic folgendermaßen angleichen:

```
1000 LET p=IN 252: LET a=USR 32871: OUT 252,p:
CLS : IF a>=1060 THEN LOAD "AUTO*"
1003 LET b=DPEEK 51760: GO SUB a: GO TO 1000
1010 DEVICE d2: GO SUB 4000: DIR 2: INPUT "File
name:";a$: LOAD a$ CODE 114688: RETURN
1015 DEVICE d1: DIR 1: INPUT "File name:";a$:
LOAD a$ CODE 114688: RETURN
1020 DEVICE d2: GO SUB 4000: GO SUB 3000: SAVE
a$ CODE 114688,b: RETURN
1025 DEVICE d1: GO SUB 3000: SAVE a$ CODE
114688,b: RETURN
1030 DEVICE d2: DIR 2: GO SUB 3000: VERIFY a$
CODE 114688,b: RETURN
1035 DEVICE d1: DIR 1: GO SUB 3000: VERIFY a$
CODE 114688,b: RETURN
1040 DEVICE d2: GO SUB 4000: DIR 2: PAUSE :
RETURN
1050 INPUT "Drive 1/2: ";d: DEVICE dd: DIR d:
INPUT "File name:";a$: ERASE a$: RETURN
2000 RETURN
3000 INPUT "File name:";a$: LET pa=114688+b-51:
POKE pa+20,MEM$ (pa TO pa+19)+CHR$ 255: RETURN
4000 INPUT "Record Nr.: ";re: RECORD re: RETURN
9800 CLEAR 32511: PRINT "FLASH Disc version 1.1
(MOUSE)": LOAD "FLASH1"CODE : LOAD "FLASH2"
CODE 6*16384+7008: CLEAR
9840 RUN
9998 DEVICE d2: SAVE "FLASH" LINE 9800: SAVE
"FLASH1" CODE 32768,57344: SAVE "FLASH2" CODE
6*16384+7008,42000: RETURN
```

Wer jetzt noch das Codefile "FLASH1" mit POKE 69481,"DRV1" und POKE 69497,"DRV_2" ändert und mit SAVE"d2:FLASH1"CODE 32768,57344 sichert, der hats nun auch im Filemenu des Programms stehen. Viel Spaß damit! **Wo vom WoMo-Team**

Die neue Mitglieder-liste 1998



Was lange währt, währt endlich gut. Hier ist sie (diesmal in alphabetischer Reihenfolge): Die Liste der 105 User, die dem SPC auch in diesem Jahr weiterhin die Treue halten, die neu hinzu gekommen sind, mit denen wir Tauschabos unterhalten etc. etc. Und da es (wieder mal) leichte Probleme gab, geht unser Dank an Lothar Ebelshäuser und Peter Rennefeld für ihre schnelle und "unbürokratische" Hilfe.

1. Atari Club Köln, Arnold Aubert
   Bonner Straße 364, 50968 Köln

Nele Abels-Ludwig
  Am Mühlgraben 4, 35037 Marburg
Roland Albert
  Hinterm Forsthaus 9, 88696 Owingen
Jean Austermühle
  Sternwartstraße 69, 40223 Düsseldorf

Stefan Ballerstaller
  Berg am Laim Str. 145, 81673 München
Klaus Barth
  Spielburg 10, 30890 Barsinghausen
Rolf Baumann
  Pferdsbruchfeld 11, 50170 Kerpen
Dirk Berghöfer
  Am Kalkrain 1, 34549 Edertal-Giflitz
Peter Bergmann
  Reiherstr. 30, 68309 Mannheim
Wolfgang Berndt
  Friedberger Str. 92c, 61169 Friedberg
Hans Joachim Blume
  Kohlrauschweg 19, 60486 Frankfurt
Eduard Bröse
  Joseph-Schmidt-Str. 23, 12057 Berlin
Günter Brütting
  Waldacher Dorfstr. 34, 91278 Pottenstein

Siegfried Dikomey
  Brunnenstraße 41, 52531 Übach-Palenberg
Wilhelm Dikomey
  Mühlengasse 24, 52391 Vettweiß
Manfred Döring
  Stuttgart-Str. 15, 70734 Fellbach

Lothar und Marion Ebelshäuser
  Bergisch Gladbacher Str. 796A, 51069 Köln
Thomas Eberle
  Rainäckerstr. 4, 70794 Filderstadt
Horst Engelhardt
  Im Eisenbach 5, 35716 Dietzhölztal

Kai Fischer
  Raumer Straße 2B, 09366 Beutha
Ingolf Fitzner
  Talstraße 7, 07407 Rudolstadt

Wolfgang Gierisch
  Habichtstr. 9, 82223 Eichenau
Slawomir Grodkowski
  Wolfgang-Döring-Str. 11, 37077 Göttingen
Michael Gruschke
  Am Seeufer 73, 56235 Ransbach-Baumbach

Stephan Haller
  Schützheider Weg 32, 51465 Bergisch Gladbach
Wolfgang und Monika Haller
  Im Tannenforst 10, 51069 Köln
Herbert Hartig
  Postfach 326, 86803 Buchloe
Günter Hartwig
  Lissabonstr. 14, 37079 Göttingen
Rudolf Herzog
  Goldbornstr. 63, 51469 Bergisch Gladbach
Detlef Heuer
  Zwiebelmarkt 37, 98701 Großbreitenbach
Rupert Hoffmann
  Tulpenstr. 22, 92637 Weiden
Dieter Hucke
  Korbacher Str. 241, 34132 Kassel
Scott-Falk Hühn
  Offenhainer Straße 4, 99610 Sömmerda

Christoph Idstein
  Bahnhofstraße 116, 55218 Ingelheim

Claus Jahn
  Felsenstraße 12, 36266 Heringen

Nico Kaiser
  Knebelstr. 5, 98693 Ilmenau
Bernd Kalla
  Robert-Koch-Str. 3, 36043 Fulda
Günter Keefer
  Erzgebirgeweg 16/1, 70736 Fellbach
Helge Keller
  Brauerstraße 7, 76137 Karlsruhe
Mustafa Knobel
  Seidelstraße 39, 13507 Berlin
Michael Kloss
  Rheinstr. 14, 56348 Kestert
Roland Kober
  Josef-Neuberger-Str. 42, 40625 Düsseldorf
Dirk Peter Küppers
  Regentenstraße 112, 41061 Mönchengladbach

Harald R. Lack
  Heidenauerstr. 5, 83064 Raubling
Peter Liebert-Adelt
  Lützowstr. 3, 38102 Braunschweig
Wolf-Dietrich Lübeck
  Jakobistr. 15, 30163 Hannover
Bernhard Lutz (b. Sprenger)
  Hammerstr. 35, 76756 Bellheim

Willi Mannertz
  Lindenstr. 12, 24223 Raisdorf
Günther Marten
  Staulinie 12, 26122 Oldenburg
Josef Menzel
  Dr. Gebauer Str. 71a, 55411 Bingen
Frank Meurer
  Schulstr. 21, 50389 Wesseling
Dirk Mayer
  Masiusstraße 9, 50827 Köln
Peter Miosga
  Holtbredde 11, 45711 Datteln
Frank-Michael Moczko
  Haselsteig 41, 12347 Berlin
Erwin Müller
  Strehlener Str. 6b, 01069 Dresden
Dieter Münz
  Iglauer Straße 44, 89518 Heidenheim

Emil Obermayr
  Teichmüllerstr. 2, 38114 Braunschweig
Christof Odenthal
  Platanenweg 8, 85609 Aschheim

Carsten Pfeil
Mittl. Landweg 226, 21035 Hamburg
Martin Pollok
Ina-Seidel-Straße 29, 40885 Ratingen

E. Reich, c/o R. Helbing
Dom.-Ringeisenweg 3, 82380 Peißenberg
Peter Rennefeld
Küpper 32, 52525 Heinsberg
Hubert Roßkamp
Kaninenberghöhe 35, 45136 Essen

Guido Schell
Auf dem Stocke 37, 32584 Löhne
Frank Schlüter
Pehlen 2B, 32108 Bad Salzuflen
Hans Schmidt
Fredersdorfer Str. 10, 10243 Berlin
Heinz Schnittker
Geiststraße 14, 59329 Wadersloh
Heinz Schober
Taubenheimer Straße 18, 01324 Dresden
Andreas Schönborn
Rüdesheimer Str. 60, 64295 Darmstadt
Wilko Schröter
W.-Barents-Str. 28, Nr. 7/4, 18106 Rostock
Bodo Schulte-Varendorff
Kiefernweg 3a, 49090 Osnabrück
Thomas Schwarz
Oberhofer Straße 82, 88096 Tettnang
Ian D. Spencer
Fichtenweg 10c, 53804 Much
Klaus-D. Stübs
Ispei 14, 58675 Hemer

Hans-Christof Tuchen
Lotzestr. 10, 12205 Berlin

Jörg Vogtschmidt
Wagnerstraße 21, 48249 Dülmen

Alexander Walz
Kastanienweg 4-6, 52074 Aachen
Paul Webranitz
Borgasse 16, 54538 Kinheim
Claus-Jörg Weiske
Veit-Stoß-Str. 2, 82256 Fürstenfeldbruck
Ingo Wesenack
Dahlmannstraße 10, 10629 Berlin
Norbert Wiedkamp
Lortzingstr. 5, 48477 Hörstel-Riesenbeck
Albert Wolter
Kieselhausenstr. 23g, 09117 Chemnitz

Dänemark:
Arne Nielsen
Chr X's vej 10st, DK-8260 Viby 7

Großbritannien:
Andy Davis
62, Tithe Barn Lane, Woodhouse, Sheffield
South Yorkshire, S13 7LN, England
Dave Fountain
11 Camel Road, Silvertown
London, E16 2DE, England
Miles Kinloch
Flat 16, 6 Drummond Street
Edinburgh, EH8 9TU, Schottland
Malcolm Mackenzie
31 Ashwood Drive, Brandlesholme, Bury
Lancs, BL8 1HF, England
Matthew Westcott
14 Daisy Mill Drive, Adlington
Chorley, Lancs, PR6 9NE, England

Lithauen:
Dalnikovas Eugenijus
Kolvariju g. 142-3, 2042 Vilnius
Sigitas Grigorius
Ateities 1-39, 2057 Vilnius

Niederlande:
Rudy Biesma
Betuwe 18, 9405 JJ Assen
Edwin Blink
Kremersheerd 63, 9737 PK Groningen
Stefan Drissen
Godevaert Montensstraat 17, 4811 PD Breda
Flora Elstrodt
Boelemahuid 151, 9736 HJ Groningen
Johan Koelman
W. van de Veldestraat 1, 5831 BW Boxmeer
Johan Koning
Mieden 6, 9866 TM Lutjegast
Roelof Koning
Selwerderstr. 26, 9717 GK Groningen
Ronald Raaijen
Hazepad 5, 8309 AX Tollebeek

Österreich:
Leszek Chmielewski Daniel
Prager Str. 92/11/12, A-1210 Wien
Georg Gojcevic
Badweg 6, A-6923 Lauterbach
Peter Meindl
Siemensgasse 3/8, A-2630 Termitz

Polen:
Ed Polinski
Boczna 27, 05-300 Minsk Maz

Rumänien:
Marin Stanculescu
SOS. Oltenitei 238, Bl. 53, Sc. 3
Et. 1, Ap. 93, Sect. 4, O.P. 8
75652 Bucuresti

Schweden:
Björn Eriksson
Axvallsvägen 54, 12150 Johanneshov

Schweiz:
Anton Arpagaus
Zwärenstr. 8, CH-4118 Rodersdorf

Spectrum-Disks auf Festplatte archivieren mit

COPYQM

Die Firma Sydex vertreibt für den PC ein Shareware-Programm namens CopyQM, daß auch für uns Spectrum-Besitzer äußerst interessant sein kann, sofern unser Spectrum mit Diskettenlaufwerken ausgestattet ist. CopyQM tut eigentlich nichts weiter, als Disketten in einem einzigen Durchgang zu kopieren, und gegebenenfalls ein Image auf der Festplatte abzulegen. Dabei können, und das ist für uns wichtig, auch Fremdformate wie z.B. das Atari-Format, das Apple II-Format und eben das +D Format gelesen werden. CopyQM verbraucht auf der PC-Seite nur sehr wenig Rechnerkapazität, ein XT mit 256K Ram, einer kleinen Festplatte und Dos 3.3 reicht schon völlig aus. Um z.B. eine +D Diskette auf die Festplatte zu bannen, ist der Befehl

```
COPYQM A: RECORD=C:blablabla
```

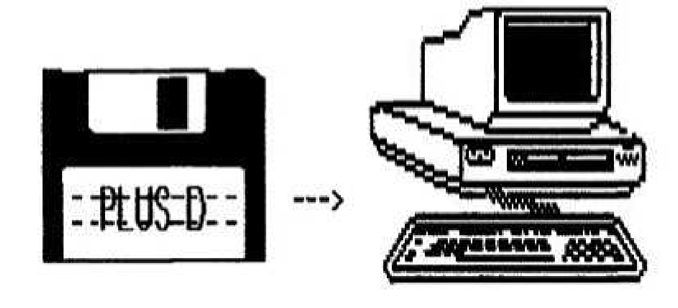
ausreichend. Um das Festplattenimage anschließend wieder auf eine für den Spectrum lesbare Diskette zu bringen reicht:

```
COPYQM A: PLAYBACK=C:blablabla
```

Dabei müssen die zu beschreibenden Disketten nicht einmal formattiert sein, das erledigt CopyQM von sich aus, und es geht nach meinen Erfahrungen dabei sehr sicher und schnell vor. Dem Anwender stehen eine Vielzahl von Switches zur Verfügung, wie z.B. SILENT, was die akustischen Signale abstellt, oder MONO, um eine schwarzweiße Bildschirmausgabe einzustellen.

Das Programm ist auf der Alchnews-Webpage erhältlich (wer keinen Internetzugang hat, wende sich an mich), und dort finden sich auch schon etliche +D Images voll mit Andy Davis' Software. Hier liegt meiner Meinung auch die wichtigste Anwendungsmöglichkeit für das Programm. Es ist damit möglich, die großen PD-Bibliotheken für den Spectrum ohne viel Aufwand im Internet verfügbar zu machen. Man muß dann keine Kapriolen schlagen, um Programme und Daten in irgendein Emulatorformat zu verwandeln, nur um es hinterher wieder für den echten Spectrum zurückzuformatieren. Ein einfacher Lesevorgang im PC, der pro Diskette ungefähr eine Minute dauert, reicht.

Ein Wermutstropfen bleibt allerdings. CopyQM kopiert nur ganze Disketten, es nicht möglich, einzelne Files aus den Images zu lesen oder in die Images zu schreiben. Das Problem ist allerdings



leicht zu umgehen, denn netterweise ist ein Imagefile schlicht und einfach ein Header von einigen Hundert Bytes, dem ein genaues 1:1 Abbild der Diskette folgt. Wenn man über das Format und die Directory-Struktur der +D Diskette Bescheid weiß, ist es also ein leichtes, eigene Software zu schreiben, die die virtuellen Sektoren des Imagefiles korrekt liest und beschreibt, und auch dafür sorgt, daß das virtuelle Directory entsprechend upgedated wird.

Ein weiterer Vorteil des Programms besteht darin, daß es sehr viel schneller Disketten formattieren kann, als der Spectrum mit dem +D. Dazu muß man nichts weiter tun, als ein Image einer frisch formattierten Diskette auf die Festplatte zu kopieren, und bei Bedarf dieses Image auf die zu formattierende Diskette zurückkopieren. Ich habe damit bis jetzt beste Erfahrungen gemacht.

SCENE MAG

Die Scene+ Diskette März/April '98

Seit Anfang dieses Jahres erscheint das Scene+ Diskettenmagazin von L.C.D. zusammen mit dem SUC Magazin. Grund genug, sich die Sache mal anzuschauen. Scene+ ist in verschiedenen Formaten erhältlich: PlusD, Opus, MB02 und Beta-Disk. Ich habe mir die PlusD-Ausgabe angeschaut.

Sobald die Diskette gebootet hat, begrüßt uns Scene+ mit einer kurzen Willkommensseite, auf der der Titel des Magazins in einer gerenderten dreidimensionalen Drehgrafik gezeigt wird. Es folgt kurz ein Scan des Raumschiffs Enterprise, den man sich aber gar nicht richtig angucken kann, da sofort das Menü nachgeladen wird. Über die Tasten A-H lassen sich die verschiedenen Programmpunkte anwählen, ein roter Balken öffnet sich dann und man kann per Cursor Unterpunkte anwählen. Es finden sich hauptsächlich Textfiles, Grafiken und Programme. Die Textfiles werden mit einem 64-Zeichenreader dargestellt, für die Bilder gibt es eine hübsche Slideshotroutine, aus der heraus man die Bilder

auch abspeichern kann. Die Programme lassen sich in den allermeisten Fällen durch Tastendruck starten (außer man hat USR 0 nicht eingeschaltet, wenn das verlangt wird) und werden von einem kurzen beschreibenden Text begleitet. Was ist in der März/Aprilausgabe enthalten?

Scene+ enthält einen umfangreichen redaktionellen Teil, in dem LCD nicht nur über die Pentagon-Schaltpläne schreibt, die er kürzlich erhalten hat, über seine Übersetzung des Spiels "Scorpions" und über ein privat entwickeltes Disketteninterface, sondern der seltsamerweise auch eine Besprechung der TV-Serie "Liebling, ich habe die Kinder geschrumpft" enthält, die vielleicht, ähm, einen Tick zu gut ausgefallen ist, aber das ist ja persönliche Geschmackssache. Des weiteren gibt es einen vollständigen Index der bisher erschienenen Scene+ Disketten, der in Artikel, Spiele, Demos, Anwendungsprogramme und Extras geteilt ist. Das ganze ist sehr praktisch, wenn man Programme oder Texte wiederfinden will, denn wenn man sämtliche paarundzwanzig Disketten des Magazins in seiner Diskettenbox liegen hat, wird es mit dem Suchen schwierig. Zwei Teile einer offensichtlich vielteiligen Science-Fiction Serie namens "Vortex" folgen. Es ist ein wenig schwierig, dieser Geschichte zu folgen - es ist nicht gerade Asimov, aber wer von uns ist schon Asimov?

Zwei Artikel haben da schon mehr mit dem Spectrum zu tun. Der erste Text ist eigentlich weniger ein Artikel, als vielmehr eine Auflistung von Z80-Mnemonics. Zuerst war ich mir nicht ganz sicher, ob noch eine weitere Opcodeliste tatsächlich von Nutzen wäre, aber diese Liste ist insofern anders, als daß sie die Opcodes in dezimaler Form angibt. Ich halte es zwar aus verschiedenen Gründen für praktischer, beim Programmieren in Maschinensprache den Hexadezimalcode zu verwenden, aber für Basicprogrammierer, die per Hand assemblieren und Data-Zeilen verwenden ist diese Liste außerordentlich hilfreich. Der zweite Artikel befaßt sich mit der Interlace-Programmierung. Es wird nicht nur beschrieben, wie die Sache funktioniert, sondern auf der Diskette finden sich auch ein Utility zur Erzeugung von interruptgesteuerten Interlacebildern und eine kurze Demo. Artikel und Routinen sind sehr interessant aber im Gegensatz zu LCD glaube ich, daß Interlaces - also zwei Bilder, die durch schnelles Hin- und Herschalten zu einem Bild verschmelzen - wenig praktische Anwendungsmöglichkeiten haben. Wenn es eine Methode gibt, die Textdarstellung des Spectrums NICHT zu verbessern, dann sind das Interlaces! Ein Textviewer - wohlmöglich noch in 64 Zeichen - im Interlacemodus würde den Verbrauch von Aspirin ins Astronomische steigern. Auch die wild flackernden Demobilder konnten mich nicht so recht überzeugen. Am besten hat mir noch das Bild von Captain Kirk gefallen.

Die Scene+ enthält jede Menge Bilder, allesamt schwarz-weiße Scans. Zum größten Teil sind dies Scans, die LCD aus japanischen Manga-Comics kopiert hat, zum Teil aber auch Tierbilder. Das dicke fette Krokodil hat mir dabei am besten gefallen. Die Manga-Scans sind zum Teil etwas schief und manchmal finden sich auf den Gesichtern die putzigen Schachbrettmuster, die immer dann entstehen, wenn man ein Raster verkleinert. Ich weiß ja nicht, welchen Scanner LCD benutzt und wie die Vorlagen original aussehen, aber um so einen Effekt zu vermeiden, ist immer ratsam, zuerst einen Farbscan zu machen, diesen auf die gewünschte Größe konvertieren und erst zuletzt die Farben in ein Raster zu wandeln.

Auf der Diskette finden sich drei Spiele: Mega Xonix, "eine Mischung aus Zolyx, Pyromania und Bomberman", Scorpions, eine Art Action-Strategiespiel, in dem es darum geht, außerirdische Angreifer, die sogennanten "Scorps", zu vertreiben, und "Nebula", ein abstraktes Science-Fiction Strategiespiel. An Demos gibt es das russische "Excess 128", das zum Teil nur auf dem Pentagon läuft, und die "E96-Interrupt Songs", ein Sammlung von Liedern, die sich zum Teil richtig gut anhören. Es war mir nicht möglich, den "X-Tracker Player" auf meinem +2 zu starten, USR 0 Mode oder nicht. Ich habe es zwar geschafft die grafische Oberfläche zu laden, aber bei der ersten Aktion scheint das Programm haltlos abzustürzen.

Zusammenfassend kann man sagen, daß das Scene+ Magazin einen recht guten Eindruck bei mir hinterlassen hat. Die Benutzeroberfläche ist gut durchdacht und nach kurzer Eingewöhnungszeit leicht zu bedienen. Es macht Spaß, sich mit einer guten Tasse Tee hinzusetzen, und sich die Demos anzusehen und die Artikel zu lesen. Vom Niveau her erreicht das Diskmag zwar nicht unbedingt Andy Davis "Alchnews", aber dieses Magazin besteht ja auch schon sehr lange, so daß sich dort eine ganz andere Routine entwickelt hat.

Nele Abels-Ludwig, Am Mühlgraben 4
35037 Marburg, Tel. 06421/210272
e-mail: abels@stud-mailer.uni-marburg.de

MOON MAGIC

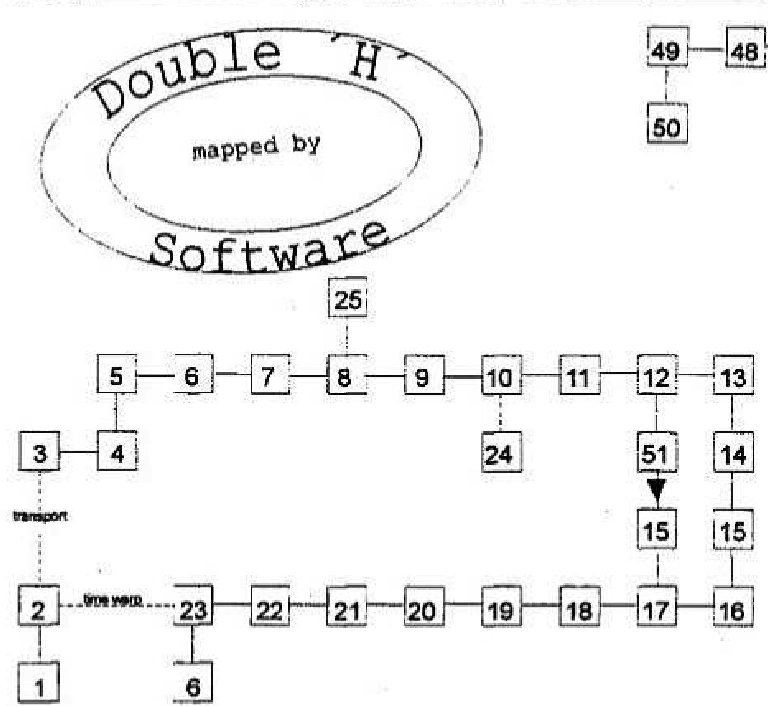
Seid gegrüßt Abenteurer!!

Wer sich mit dem Spectrum und hier insbesondere mit Adventureprogrammen beschäftigt, der kennt sicherlich den Namen "HEDGEHOG". Nun, dahinter verbirgt sich niemand anderer als A. J. Remic, der für den Spectrum eine Menge an Programmen geschrieben hat, darunter auch diverse Adventure. Einem dieser Programme wollen wir uns heute widmen, und zwar "MOON MAGIC". Wie der Name schon vermuten läßt, spielt sich die ganze Sache im Weltraum ab, zählt also zum Genre der Spaceadventure. Das Programm wurde Anfang der neunziger Jahre geschrieben und weißt unseres Erachtens einen mittleren Schwierigkeitsgrad auf, d. h. es ist weniger umfangreich wie Programme aus dem Hause Level 9 und auch die Aufgabenstellung ist einfacher gehalten. Trotzdem ist das Adventure recht gut zu spielen. Ziel ist es, mit unserem Raumschiff FIG (vermutlich eine Abkürzung für Fast Intergalactic Glider) vom Mond wegzukommen, auf dem wir festsitzen. Denn auch in Zeiten der Raumfahrt mangelt es anscheinend des öfteren an Treibstoff. Also machen wir uns auf die Suche nach einer "Tankstelle" damit wir unser Schiff wieder flott bekommen. Dabei durchqueren wir folgende Locations, die ihr auch auf dem beiliegenden Plan ersehen könnt:

01) In the starship FIG / terminal transport card
02) In the transport computer room / psychadelic folder
03) On the moon's surface A / elongated cardboard tube
04) On the moon's surface B
05) On the moon's surface C
06) On the moon's surface D / uncharged laser
07) On the moon's surface E
08) In a small trench
09) In a steep sided gully
10) In the deepest part of the gully
11) At the end of the steep sided gully
12) On the moon's surface F
13) In a small abandoned cave
14) In a dried up underground river
15) In the river trench
16) At the end of a dried up river trench
17) At the end of a river trench
18) In the tributary cave
19) In a long low ceilinged tributary cave
20) In the low ceilinged passage
21) In a large cavern A
22) In a large cavern B
23) In the connecting chamber
24) In a big cavern / officers valid I. D. strip
25) In an airlock
26) In the spaceport transport bay
27) In a droids sleeping quarters / stick of dynamite
28) In a big council room
29) In a mechanical reservice station
30) In the spaceport's telefunk room / match
31) In an oil servicing room / fuse for dynamite
32) In an interstellar galactic chamber / pin
33) In the refreshments room
34) In the oil drinks corridor
35) At the end of the oil drinks room / chain
36) In a big supplies store
37) At the end of the supplies store / empty hollow tin
38) In a long ventilation shaft
39) In the long low metal shaft
40) At a bend in the long ventilation shaft / voodoo doll
41) In another bend in the ventilation shaft
42) At a dead end in the ventilation shaft / molecule transporter
43) In the molecule transporter
44) In a basin shaft
45) In an interconnecting level / code key
46) In a travel tube
47) Between the tracks in the transport shaft
48) In the transport shaft
49) In a slight bend in the transport shaft
50) At the end of the transport shaft / tube of nuclear fuel
51) In a dried up river trench

Soviel zu den Locations des Planes. Kommen wir jetzt zur Lösung dieses Adventures. Wir befinden uns an unserer Startposition an Bord des Raumschiffes FIG.....

Take card, examine card, N (wir befinden uns im Transporterraum - fast wie bei Raumschiff Enterprise!), transport (und schon werden wir auf den Planeten gebeamt). Wir sind jetzt auf der Mondoberfläche....E, N, E, take laser, E, E, E, E, E, E (in a small abandoned cave), S, S, S (end of the dried up river trench), W, W, W, W, W, W, W



(in a chamber), recharge laser, S, E, E, E, E (hier gibt es einen kleinen Eingang in Richtung Süden), fire laser (wir erreichen eine große Kammer), drop laser (wird nicht mehr benötigt), take strip, examine strip (damit öffnet man Luftschleusen), N, W, W, N (wir sind in der Luftschleuse), insert strip (wir gelangen in den Transporterraum der Raumstation wo uns ein durstiger Androide nicht nach Osten lassen will), drop strip, W (in den Schlafräumen), take dynamite, N, N, N (die Funkzentrale), take match, E (der Ölservice Raum), take fuse, S (wir sind in der interstellaren galaktischen Kammer), take pin, S, W (wir sind im Versammlungsraum),(light match), light fuse (das klappt aber nur, wenn wir auch das Dynamit dabei haben. Damit schaffen wir dann eine Möglichkeit um nach Westen zu gelangen und zwar in den Korridor mit den Öldrinks -igitt!?!), drop match, drop fuse (merkwürdigerweise haben wir den Zünder noch), W, S, S, take tin, examine tin, N, N, take chain, pull chain (Jetzt ist unser Behälter voll mit Öl), S, E, E (wir befinden uns wieder im Transporterraum der Raumstation), give can (der durstige Androide freut sich und wir gelangen in den Ventilationsschacht), E, E, take doll, N, E, N (Molekular transporter), E, stick pin (wir stechen mit der Nadel in die Puppe und der Androide stirbt - scheint irgendwie mit Voodoo zu tun zu haben), take key, W (wieder im Tranporterraum), N, N, N, W, insert key (eine Geheimtür öffnet sich), W, W, W, S, take fuel, N, E, E, E, E, S, S, S, E, S, S, E, E, E, E, S, S, S, W, W, W, W, W, W, W (wir fallen in einen Timewarp und befinden uns wieder an Bord unseres Raumschiffes), S, refuel ship......

Das wars. Moon Magic ist gelöst. Wir hoffen ihr hattet wieder Spaß beim Durcharbeiten. Bis demnächst hier mit einem neuen Abenteuer......

Harald R. Lack, Heidenauer Str. 5, 83064 Raubling
Hub. Kracher, Schulweg 6, 83064 Großholzhausen



Datei-verwaltungs-system (2)

Im Teil 1 wurden grundsätzliche Ausführungen zum Dateiverwaltungssystem gemacht. Im Teil 2 geht es um das Anlegen des VOL1-Kennsatzes auf der Diskette. Vorausgesetzt wird eine mittels des Plus-D formatierte Diskette. Vor dem Laden des BASIC-Programmes KATVHGENER ist mit CLEAR 63999 ein Freiraum für den Pufferbereich zu schaffen, in dem der einzulesende bzw. auszugebende Sektor seinen Platz im Speicher hat. Sonst erfolgt die Fehlermeldung "RAMTOP nicht neu gesetzt". Wird eine andere Pufferadresse als 64000 angegeben, erfolgt die Fehlermeldung "Pufferadresse unerlaubt!". Nach der Eingabe der Pufferadrese wird mit der Meldung "Disk. Einl. -> Taste!" zum Einlegen der vorgesehenen Diskette in eines der beiden Laufwerke aufgefordert. Nach

```
                 VOL1GENERI
Laufwerk:        ==========          PUFFER-
VOL1 generieren[j/n]:                ADRESSE
----
VOL1 existiert schon:

Diskname alt:              Diskettenname
Diskname neu:              aendern[j/n]:

Freie HDR1's:              Datum        Zeit
-------------              -----        ----
Ist -Datum&Zeit            .  .          .
Gen.-Datum&Zeit            .  .          .
Akt.-Datum&Zeit            .  .          .

Naechste freie Datenspur:
Naechster freier Datensektor:
Naechster freier HDR1-Sektor:
Naechste freie HDR1-S.-Posit.:
Freie Datensektoren:

FEHLER:
```

Abb.1: Vor der ersten Eingabe

```
          VOL1GENERI
Laufwerk:2 ==========      PUFFER-
VOL1 generieren[j/n]:  j  ADRESSE
----                       : 64000
VOL1 existiert schon: n

Diskname alt: 023 Diskettenname
Diskname neu: 023 aendern[j/n]:n

Freie HDR1's:        Datum     Zeit
-------------        -----     ----
Ist -Datum&Zeit 07.05.1998 13.45
Gen.-Datum&Zeit 07.05.1998 13.45
Akt.-Datum&Zeit 07.05.1998 13.45

Naechste freie Datenspur: 001
Naechster freier Datensektor: 01
Naechster freier HDR1-Sektor: 02
Naechste freie HDR1-S.-Posit.: 1
Freie Datensektoren: 1590
            Disk. einl. -> Taste!
FEHLER:
```

Abb. 2: Bild gefüllt mit Eingabe und automatisch

Abb. 3: Ende der Eingabe, weil HDR1-Erzeugung verneint

```
          HDR1GENERI
Laufwerk:  ==========      PUFFER-
HDR1 generieren[j/n]:  n  ADRESSE
----                       :
Diskettenname:            Alt  Neu
                          ---  ---
Next free DATA-Track:
Next free DATA-Sektor:
Next free DATA-Position:
Next free HDR1-Sektor:
Next free HDR1-S.-Posit.:
Freie HDR1-Positionen

Datum:    .  .       Uhrzeit:  .
------               --------Satz-
Dateilaenge:         laenge: -----
------------         -------

Dateiname:
----------

FEHLER:
```

dem Drücken einer beliebigen Taste geht es weiter. Das erste Bild dient zur Eingabe des VOL1-Kennsatzes (Abb. 1).

Bei einer neu initialisierten Diskette werden durch Test des 1. Sektors der Spur 0 automatisch "VOL1 generieren" auf "j" und VOL1 existiert schon" auf "n" gesetzt. Bei der Frage nach dem Diskettennamen ist eine Zahl von 0 bis 999 einzugeben. Diese Nummer identifiziert die Diskette. Sie wird sowohl bei "Diskname alt" als auch bei "Diskname neu" eingetragen. "Diskettenname aendern" wird automatisch auf "n" gesetzt. Das Datum ist als 8-stellige Zahl (String) in der Form 2-stelliger Tag, 2-stelliger Monat und 4-stelliges Jahr einzugeben. Die Uhrzeit wird als 4-stellige Zahl (String) in der Form 2-stellige Stunde und 2-stellige Minute eingegeben. Es erscheinen sonst BASIC-Fehler 1 bzw. 2. Beide Werte werden syntaktisch überprüft und über eine Fehlermeldung wird zur Wiederholung der Eingabe aufgefordert. Auch die Prüfung auf Schaltjahre ist vorgesehen. Die weiteren Felder in diesem Bild werden automatisch mit den jetzt aktuellen Werten gefüllt. Die Diskette hat insgesamt 1600 Sektoren. Die Spur 0 mit ihren 10 Sektoren stellt das VTOC (Verzeichnis der Diskette) dar. Somit verbleiben noch 1590 Sektoren für die Speicherung der Daten der Dateien (Abb. 2 und 3).

Zum Schluß der VOL1-Generierung erfolgt noch die Angabe des Aufbaus des VOL1-Kennsatzes (VOL1-Sektor) in Tabellenform. Seine Kenntnis ermöglicht es, im Fehlerfalle gezielte Korrekturen vornehmen zu können.

Feld-Nr.	Adresse Absolut	Adresse Relativ	Länge	Attribut	Name	Bedeutung
1	1	0	4	Character	VOL1	Datenträgerkennung
2	5	4	3	"	Vdn	Datenträgername
3	8	7	8	"	Gdatum	Generierungsdatum
4	16	15	4	"	Gzeit	Generierungszeit
5	20	19	2	"	Nesek	Nächster freier HDR1-Sektor
6	22	21	1	"	Nesekp	Nächste freie HDR1-Position im HDR1-S.
7	23	22	2	"	Frehp	Anzahl noch freier HDR1-Positionen
8	25	24	3	"	Neftrk	Nächste freie Datenspur
9	28	27	2	"	Nefsek	Nächster freier Datensektor
10	30	29	4	"	Fresek	Anzahl noch freier Datensektoren
11	34	33	8	"	Adatum	Aktualisierungsdatum
12	42	41	4	"	Azeiz	Aktualisierungsuhrzeit
13	46	45	3	"	LEER	Nicht belegt
14	49	48				

Die Belegung des VOL1-Sektors ab Feld 14 wird im Zusammenhang mit dem Dateilöschprogramm erklärt (Fortsetzung folgt).

Erwin Müller, Strehlener Straße 6B, 01069 Dresden